Graph-based Recommendation Systems: Comparison Analysis Between Traditional Techniques and Graph Based Techniques

Project Proposal

Greshma BABU
DSBA CentraleSupélec
Gif-sur-Yvette, France
greshma.babu@student-cs.fr

Wanci HE
DSBA CentraleSupélec
Gif-sur-Yvette, France
wanci.he@student-cs.fr

Raghuwansh RAJ
DSBA CentraleSupélec
Gif-sur-Yvette, France
raghuwansh.raj@student-cs.fr

Vivian KOUTROUMANI
DSBA CentraleSupélec
Gif-sur-Yvette, France
vivian.koutroumani@student-cs.fr

Keywords: Neural Embedding; Random Walk; Graph Neural Networks; Bipartite Graph

1 Motivation and Problem Definition

Recommendation systems include techniques and algorithms which are able to suggest "relevant" items to users. The suggested items should be the most relevant possible item for the user, so that the user would be more likely to choose those items: Amazon products, news articles, Youtube videos etc.

Focusing on an amazon products dataset, items are ranked according to their relevancy, and the most relevant ones are shown to the user. Our recommendation system should be able to determine the relevancy, based on historical data. If a customer has recently bought a coffee machine, then the system should start recommending coffee capsules and other products related to coffee machine.

Recommendation systems' traditional approaches are divided into two main categories: collaborative filtering and content-based systems[4]. Traditional algorithms based on collaborative filtering require an up-to-date dataset of users and their preferences, which is difficult to gather for huge database of items. Also, content-based approach suffers from the complex computation of similarity among items[2]. More recently, neural embedding, especially word2vec, has shown effective in the recommendation system[1]. Over the last few years, Graph Learning based Recommendation Systems have been developed on graphs where the important objects, e.g., customers, products, and attributes, are either explicitly or implicitly connected[5]. A graph-based recommendation system using neural embedding to capture similarity in a sparse dataset seem sensible and thus worth exploring. Given the Amazon product co-purchasing network metadata, we use neural embedding to predict the co-purchase bipartite graph of items and baskets of customers. We chose to work on this project because we think that recommendation systems are gaining popularity among platforms, now that web is the dominant type of advertising. We consider it as an extremely interesting application of network science and hence, would like to explore and experiment with it.

2 Methodology

We used the Amazon Product Co-Purchasing Network metadata provided by SNAP. This dataset holds a variety of product purchase metadata of 721,342 items, including product categories, reviews, and aggregated, filtered lists of co-purchased products ranging from books, music CDs, DVDs, and VHS video tapes.This dataset does not contain any explicit definition of nodes and edges, and for the purpose of this project we have decided to generate two independent bipartite graphs. we compare different algorithms for recommendation systems.

2.1 Collaborative Filtering

Collaborative filtering (CF) has been a well-known algorithm for recommendation system. CF can be divided into two subcategories: memory-based collaborative filtering and model-based collaborative filtering. For the memory-based CF, there are two approaches, one based on items, and the other based on users. The algorithm draws inferences about

ESSEC-CentralSupélec, Feb 2022, Paris, France

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

the relationship between different products based on which ones are purchased together. For the item-based collaborative filtering, we directly compare the similarities between the items by calculating how often the items appeared in the same shopping cart

2.2 Random Walk

An alternative method to heuristically determine similarity between two items is to perform random walks on two items and find the Jaccard similarity between all the item nodes both walks have landed on.

2.3 Neural Embedding Candidate Generation

Neural embeddings, specifically word2vec models, have shown to be very effective in multiple industries for generating a compact representation of objects in high dimensional latent space. In most cases, these models are trained to minimize the cosine similarity between an input object and its target object. we explore the effectiveness of Neural Embeddings as a means of co-purchase candidate generation. More specifically, given a co-purchase bipartite graph of baskets and items where each basket holds a flattened representation of multiple user shopping-cart sessions, we analyze the explorative properties of Neural embeddings learned per item.

2.4 Recommendation in Bipartite Graphs

This method talks about a kernel-based recommendation approach that indirectly inspects customers and items related to user-item pair to predict whether an edge may exist between them.It has set of users U and a set of items I as nodes, and transaction was represented as an intersection graph where an edge between the two signified purchase of the product. The graph kernel was defined on the user-item pairs context, and thus item recommendation problem is equivalent to predicting a link between user-item based on graphical analysis.

2.5 Graph Neural Networks for recommendations systems

Since GNN has the ability of high-order relationship modeling by aggregating information from neighbor nodes, after fusing multiple sequences into one graph, it can learn representations of both users and items in different sequences, which can't be accomplished by Markov model or recurrent neural network.We propose a simple method that directly converts the sequence information into directed edges on the graph and then uses GNN to learn representations.We construct an item-item graph,where the edges of the item-item graph indicate co-occurrence in a sequence, with edge weights assigned according to the number of occurrences. The representations learned by GNN are used in the final recommendation through the recurrent neural network.

3 Evaluation

In the paper by Se Won Jang, Simon Kim, JeongWoo Ha titled 'Graph-based Recommendation Systems: Comparison Analysis between Traditional Clustering Techniques and Neural Embedding'[3], they compare Collaborative Filtering, Neural Embeddings Candidate Generation and Random Walk. We hope in our project to experiment and compare these models and more.

For evaluation, this paper recommends making use of two important metrics - the distance and reachability metrics. We shall measure the distance between each query item and candidate item and compare for different models. The higher the maximum distance and standard deviation, the more items that the model could potentially recommend, i.e, the model can capture similarities that may seem distant.

The second metric, reachability, is measured to test the hypothesis that neural models can capture co-purchasability of items that are not connected on the graph generated from a limited dataset.

To make sense of our recommendations, we shall also use examples, to see qualitatively whether the recommendations actually do capture similarities or not.

References

[1] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM conference on recommender systems*. 191–198.

[2] Lubos Demovic, Eduard Fritscher, Jakub Kriz, Ondrej Kuzmik, Ondrej Proksa, Diana Vandlikova, Dusan Zelenik, and Maria Bielikova. 2013. Movie recommendation based on graph traversal algorithms. In *2013 24th International Workshop on Database and Expert Systems Applications*. IEEE, 152–156.

[3] Se Won Jang, Simon Kim, and J Ha. 2007. Graph-based recommendation systems: Comparison analysis between traditional clustering techniques and neural embedding.

[4] Francesco Ricci, Lior Rokach, and Bracha Shapira. 2011. Introduction to Recommender Systems Handbook. In *Recommender Systems Handbook*.

[5] Ivan F. Videla-Cavieres and Sebastián A. Ríos. 2014. Extending market basket analysis with graph mining techniques: A real case. *Expert Syst. Appl.* 41 (2014), 1928–1936.